Amendments to the claims,
Listing of all claims pursuant to 37 CFR 1.121(c)

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently amended) A method for securing a program comprised of a plurality of interoperable components, the method comprising:

extracting export information about a function of a first component of the program that is callable by at least one other component of the program;

securing the extracted export information;

in response to an attempt by a second component ~~of the program~~ to invoke the function of the first component, validating authenticity of the second component; ~~and~~

if the authenticity of the second component is validated, providing access to the function of the first component using the secured extracted export information; and

otherwise, blocking the attempt by the second component to invoke the function.

2. (Currently amended) The method of claim 1, further comprising:

generating a signature for at least one other component of the program authorized to call the function of the first ~~the second~~ component, so as to enable authentication of said one other ~~the second~~ component.

3. (Original) The method of claim 2, wherein said step of generating a signature includes generating a selected one of an Authenticode signature and an MD5 message digest.

4. (Currently amended) The method of claim 2, wherein said step of generating a signature includes generating a hash of said one other ~~the second~~ component and encrypting the hash with a private key.

5. (Original) The method of claim 4, wherein said validating step includes decrypting the hash with a public key and comparing the hash to a known value.

6. (Currently amended) The method of claim 1, wherein said extracting step includes removing the ~~extracting~~ export information from an export table of the first component.

7. (Original) The method of claim 1, wherein said extracting step includes removing the function name from an export table of the first component.

8. (Original) The method of claim 1, wherein said securing step includes obscuring the function name.

9. (Currently amended) The method of claim 1, wherein said securing step includes creating a secure export table for securing the extracted export information.

10. (Original) The method of claim 1, wherein said providing step includes routing a call by the second component to the function of the first component.

11. (Original) The method of claim 1, wherein said providing step includes returning an address of the function of the first component to the second component.

12. (Currently amended) The method of claim 1, wherein said extracting step includes extracting export information about a function of the first program specified by a user.

13. (Original) A computer-readable medium having processor-executable instructions for performing the method of claim 1.

14. (Currently amended) The method of claim 1, further comprising:

providing a~~A~~ downloadable set of processor-executable instructions for performing the method of claim 1.

15. (Original) A method for securing a program comprised of a plurality of

modules, at least one of the modules having export information for allowing other modules to invoke its program code, the method comprising:

generating signatures for at least some of the program's modules;

as the program is loaded, validating said signatures so as to verify authenticity of respective modules of the program;

for each module having program code that may be invoked by another module, removing that module's export information;

securely storing any removed export information;

for each module having its export information removed, blocking any attempt from another module to invoke its program code if the other module cannot be authenticated; and

if the other module is authenticated, allowing the attempt to proceed using the securely stored export information.

16. (Original) The method of claim 15, wherein said generating step includes generating a selected one of an Authenticode signature and an MD5 message digest.

17. (Original) The method of claim 15, wherein said generating step includes generating a hash of a module and encrypting the hash with a private key.

18. (Original) The method of claim 17, wherein said validating step includes decrypting the hash with a public key and comparing the hash to a known value.

19. (Original) The method of claim 15, further comprising:

providing a security module for validating authenticity of a module.

20. (Original) The method of claim 19, wherein the security module includes instructions causing the security module to be initialized before other modules of the program.

21. (Original) The method of claim 19, wherein an attempt to invoke a module

having its export information removed is routed to the security module.

22. (Original) The method of claim 21, wherein the security module allows the attempt to proceed if the other module making the attempt is authenticated.

23. (Original) The method of claim 15, wherein said allowing step includes returning an address of program code of the module having its export information removed if the other module is authenticated.

24. (Original) The method of claim 15, wherein said removing step includes removing export information for a particular module specified by a user.

25. (Original) The method of claim 15, wherein said removing step includes removing information from an export table.

26. (Original) The method of claim 15, wherein said securely storing step includes obscuring removed export information.

27. (Original) The method of claim 15, further comprising:

in response to an attempt to invoke program code of a given module, verifying authenticity of the given module and blocking the attempt if the given module cannot be authenticated.

28. (Original) The method of claim 15, further comprising:

after allowing the attempt to proceed, providing for subsequent attempts by the other module to invoke the program code to directly invoke the program code.

29. (Original) A computer-readable medium having processor-executable instructions for performing the method of claim 15.

30. (Currently amended) The method of claim 15, further comprising:

providing a~~A~~ downloadable set of processor-executable instructions for performing the method of claim 15.

31. (Currently amended) A system for securing a program comprised of a plurality of interoperable components, the system comprising:

a module for extracting export information about a function of a first component of the program that is callable by at least one other component of the program;

a module for securing the extracted export information;

a validation module for validating authenticity of a second component attempting to obtain export information to invoke the function of the first component; and

a security module for blocking the attempt to invoke the function of the first component if the second component cannot be authenticated.

32. (Original) The system of claim 31, wherein the validation module validates authenticity of the second component based on examining a digital signature of the second component.

33. (Currently amended) The system of claim 31, further comprising:

a module for generating a signature for at least some components of the program, so as to enable authentication of said at least some components~~modules~~.

34. (Original) The system of claim 33, wherein the module for generating generates a selected one of an Authenticode signature and an MD5 message digest.

35. (Original) The system of claim 33, wherein the module for generating generates a hash of a module and encrypts the hash with a private key.

36. (Original) The system of claim 35, wherein the validation module decrypts the hash with a public key and compares the hash to a known value.

37. (Currently amended) The system of claim 31, wherein the security module

routes the attempt to invoke the function to ~~of~~ the first module using the extracted export information if the second module is authenticated.

38. (Original) The system of claim 31, wherein the security module returns an address of the function of the first module if the second module is authenticated.

39. (Original) The system of claim 31, wherein the module for extracting removes an export table entry for the function of the first module.

40. (Currently amended) The system of claim 31, wherein the module for securing creates a secure export table including the extracted export information.

41. (Original) The system of claim 40, wherein the secure export table is created without using a clear text name for the function of the first module.

42. (Original) The system of claim 40, wherein the module for securing obscures function names in the secure export table.

43. (Original) The system of claim 31, wherein the security module inserts executable code into the second module during initialization of the second module so as to direct an attempt by the second module to invoke the function of the first module to the security module.

44. (Original) The system of claim 43, wherein the security module modifies the executable code included in the second module if the second module is authenticated so as to enable the second module to directly invoke the function of the first module.

45. (Currently amended) A method for securing an exported function of a program, the method comprising:

extracting export information about the exported function of the program;

securing the extracted export information;

intercepting an attempt to access the exported function by an importer;

authenticating the importer for determining whether to permit access to the exported function; ~~and~~

if the importer is authenticated, providing access to the exported function based on the secured extracted export information; and

otherwise, blocking access to the exported function.

46. (Original) The method of claim 45, wherein the importer comprises another module of the program.

47. (Original) The method of claim 45, wherein the importer comprises another program.

48. (Canceled)

49. (Original) The method of claim 45, wherein said authenticating step includes authenticating the importer based on a digital signature of the importer.

50. (Original) The method of claim 45, further comprising:

generating a digital signature for at least some executable modules of the program, so as to enable authentication of said at least some executable modules.

51. (Original) The method of claim 50, wherein said generating step includes generating a selected one of an Authenticode signature and an MD5 message digest.

52. (Original) The method of claim 50, wherein said authenticating step includes validating digital signature of the importer.

53. (Original) The method of claim 45, further comprising:

authenticating a program module including the exported function before providing access to the exported function.

54. (Original) The method of claim 45, wherein said providing step includes routing a call by the importer to the exported function.

55. (Original) The method of claim 45, wherein said providing step includes returning an address of the exported function to the importer.

56. (Original) The method of claim 45, wherein said extracting step includes removing an export table entry for the exported function.

57. (Original) The method of claim 45, wherein said securing step includes obscuring the exported function name.

58. (Original) The method of claim 45, wherein said securing step includes creating a secure export table based on the extracted export information.

59. (Original) A computer-readable medium having processor-executable instructions for performing the method of claim 45.

60. (Currently amended) The method of claim 45, further comprising:

providing a~~A~~ downloadable set of processor-executable instructions for performing the method of claim 45.